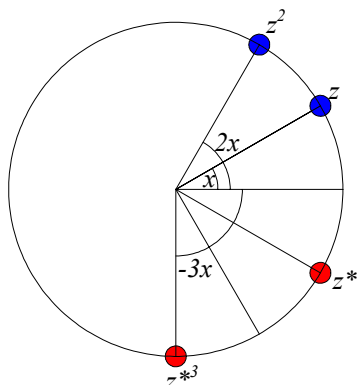


Bedeutung der diskreten und schnellen Fouriertransformationen (DFT und FFT)

FFT's sind die am häufigsten gebrauchten mathematischen Algorithmen des Alltags. Myriaden solcher Transformationen werden zum Erstellen komprimierter Musik (mp3) oder komprimierter Bilder (jpeg) durchgeführt. Bei der Schwingungsanalyse gehören sie zum Alltag des Messingenieurs („Spektralanalyse“).

Komplexe Zahlen (Wiederholung)



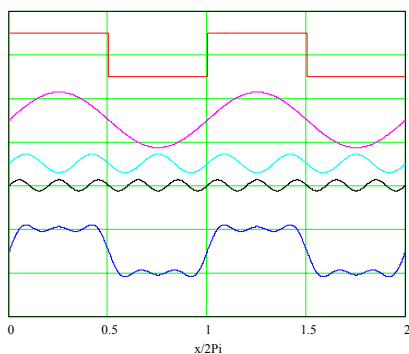
werden als Punkte in der komplexen Zahlenebene dargestellt. Hier sind vor allem komplexe Zahlen mit Betrag 1 interessant. Sie liegen auf dem Einheitskreis. Ihre Phasen nennen wir jetzt mal x . Das ist der Winkel zur Rechtsachse hin (= reelle Achse). Nach Euler kann man schreiben:

$$\begin{aligned} z &= \cos x + j \sin x = e^{jx} \\ z^* &= \cos x - j \sin x = e^{-jx} \end{aligned} \tag{1}$$

z^* ist dabei die zu z konjugiert komplexe Zahl, die durch Vorzeichenwechsel beim Imaginärteil oder durch Spiegelung an der reellen Achse entsteht.

Beim Potenzieren solcher Zahlen werden nur die Phasen vervielfacht. z^2 hat z.B. die Phase $2x$, $(z^*)^3$ die Phase $-3x$. Die Punkte drehen sich also auf dem Einheitskreis beim Potenzieren.

Komplexe Fourierkoeffizienten



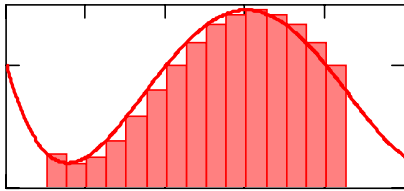
In den trigonometrischen Reihen reeller 2π -periodischer Funktionen kommen nur reelle a - und b -Koeffizienten vor:

$$\begin{aligned} f(x) &= f(x \pm n \cdot 2\pi) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \\ \left. \begin{matrix} a_n \\ b_n \end{matrix} \right\} &= \frac{1}{\pi} \int_0^{2\pi} f(x) \begin{Bmatrix} \cos nx \\ \sin nx \end{Bmatrix} dx \quad n = 0, 1, \dots \quad b_0 = 0 \end{aligned} \tag{2}$$

Aus ihnen kann man folgendermaßen komplexe Fourierkoeffizienten c konstruieren (nicht um die Studenten zu ärgern, sondern weil die Formeln viel ergonomischer werden und man sich viel Schreib- und Rechenarbeit erspart!). In einem komplexen c sind dabei die beiden reellen Zahlen a und b zusammengefasst:

$$\begin{aligned} c_n &:= \frac{a_n - j \cdot b_n}{2} \stackrel{\text{wg(2)}}{=} \frac{1}{2\pi} \int_0^{2\pi} f(x) \cdot (\cos nx - j \sin nx) dx = \\ &\stackrel{\text{wg(1)}}{=} \frac{1}{2\pi} \int_0^{2\pi} f(x) \cdot e^{-jnx} dx \quad n = 0, 1, 2, \dots \end{aligned} \tag{3}$$

Diskrete Fourier Transformation



wird die numerische Integration von Integral (3) mit Hilfe der Treppenstufenmethode genannt. Dies wird notwendig, wenn die Integrale in (2) oder (3) analytisch nicht mehr ausgewertet werden können, oder wenn von vorneherein die Funktionswerte nur punktweise vorliegen, also z.B. bei elektronischer Datenerfassung. Das Integrationsintervall $0 \dots 2\pi$ wird wie gewohnt in N gleich breite Streifen unterteilt, die Funktionswerte am linken Streifenrand sollen y_k heißen, $k = 0 \dots N-1$. N hat oft den Wert 512, 1024 usw., lässt sich also als 2er-Potenz schreiben: $N = 2^m$. Um nicht neue Buchstaben einführen zu müssen, nennen wir das Ergebnis der numerischen Integration wiederum c , behalten aber im Hinterkopf, dass es sich um eine Näherung der komplexen Fourierkoeffizienten handelt.

$$x_k := \frac{2\pi}{N} \cdot k \quad y_k := f(x_k) \quad k = 0, 1 \dots N-1$$

$$c_n = \frac{1}{2\pi} \sum_{k=0}^{N-1} f(x_k) \cdot e^{-jnx_k} \cdot \Delta x \cong \frac{1}{2\pi} \int_0^{2\pi} f(x) \cdot e^{-jnx} dx \tag{4}$$

$$c_n = \frac{1}{2\pi} \sum_{k=0}^{N-1} y_k \cdot e^{-jn \frac{2\pi}{N} \cdot k} \cdot \frac{2\pi}{N} = \frac{1}{N} \sum_{k=0}^{N-1} e^{-j \frac{2\pi}{N} n \cdot k} \cdot y_k$$

Die blaue Formel in (4) stellt genau die **Diskrete Fourier Transformation (DFT)** dar. Mithilfe des sog. „Schnörkelfaktors“ (twiddle factor im

Englischen) $w := e^{-j \frac{2\pi}{N}}$ kann die DFT in Matrixschreibweise dargestellt werden:

Beispiel $N=4$

$$w = e^{-j \frac{2\pi}{4}} = -j$$

$$\mathbf{F} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{pmatrix}$$

$$\begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{N-1} \end{pmatrix} = \frac{1}{N} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & w^1 & w^2 & \dots & w^{N-1} \\ 1 & w^2 & w^4 & \dots & w^{2N-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & w^{N-1} & w^{2N-2} & \dots & w^{(N-1)(N-1)} \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_{N-1} \end{pmatrix} \tag{5}$$

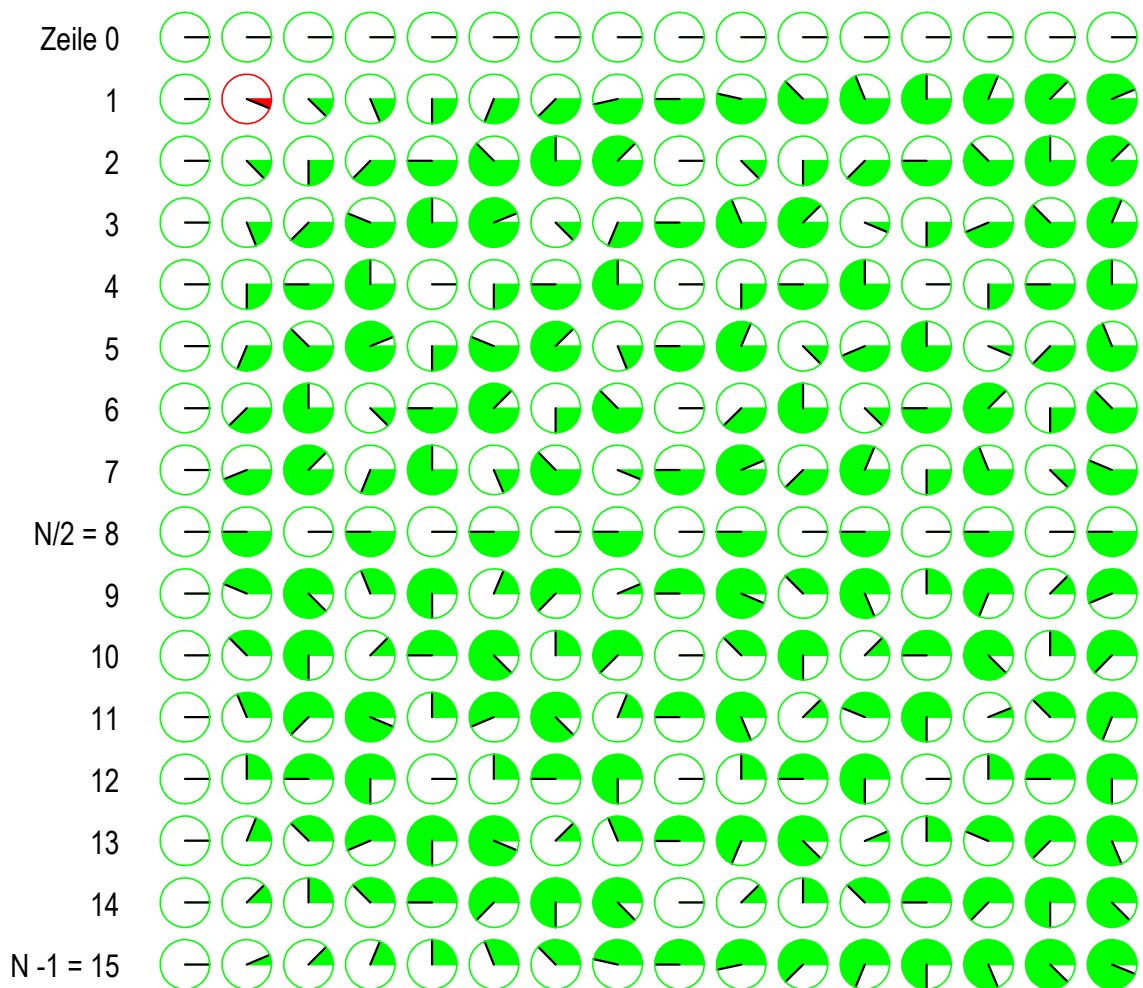
kürzer: $\vec{c} = \frac{1}{N} \cdot \mathbf{F} \cdot \vec{y}$

Matrix \mathbf{F} hat dabei die komplexen Komponenten:

$$F_{nk} = w^{n \cdot k} \quad n, k = 0, 1 \dots N-1$$

Eigentlich müsste in (5) der Index n bei den c 's bis unendlich laufen (s. Formel 3). Beim Versuch, die Matrix \mathbf{F} nach unten zu verlängern, stellt man aber fest, dass sich die Zeilen wiederholen. Zeilenindex N z.B. würde gemäß $w^{N \cdot k} = e^{-j \cdot 2\pi \cdot k} = 1$ für alle Spalten k lauter Einsen enthalten genau wie Zeilenindex 0, Zeilenindex $N+1$ ergäbe dieselben Zahlen wie Zeilenindex 1 usw. Entsprechend würde sich im Spaltenvektor c die Zahlenfolge ab c_N wiederholen. Man erkennt: **Aus N reellen Funktionswerten y_k ($k=0, 1 \dots N-1$) lassen sich nur N verschiedene komplexe Fourierkoeffizienten c_n ($n=0, 1 \dots N-1$) mittels DFT bestimmen**

Eigenschaften der DFT-Matrix F (N geradzahlig)



In Polardiagrammen dargestellt ist oben die Matrix F mit $N=16$. Zeile 0 sowie Spalte 0 bestehen immer aus lauter Einsen. Der Schnörkelfaktor w befindet sich immer an Indexposition 1,1 (rot). Mit ihm ergibt sich Zeile 1 durch Rotation um jeweils den roten Winkel ($2\pi/16 = 22,5^\circ$ im Beispiel). Die gesamte Zeile 2 ergibt sich durch Winkelverdopplung von Zeile 1, Zeile 3 durch Winkelverdreifung von Zeile 1 usw. Man erkennt:

In Zeile $c_{N/2}$ stehen 1, -1 im Wechsel. Der untere Rest (Zeile 9 .. 15) der Matrix ergibt sich durch Spiegelung der Zeilen 7 .. 1 an dieser Zeile 8. Die einzelnen Polardiagramme werden dabei auch um die reelle Achse gespiegelt. Dem entspricht die mathematische Operation konjugiert komplex, wie eingangs erwähnt.

Die gesamte Matrix ist außerdem symmetrisch (transponierte Matrix $F^T = F$).

In Formeln:

Spiegelsymmetrie
bzgl. mittlerer Zeile

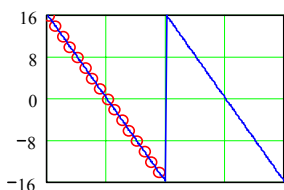
$$F_{(N-n)k}^* = F_{nk}$$

Die c 's sind damit
auch „symmetrisch“
bezüglich $c_{N/2}$!

$$c_{(N-n)}^* = \left(\frac{1}{N} \sum_k F_{(N-n)k} \cdot y_k \right)^* = \frac{1}{N} \sum_k F_{(N-n)k}^* \cdot y_k = \frac{1}{N} \sum_k F_{nk} \cdot y_k = c_n \quad (y_k \text{ und } N \text{ reell!}) \quad (6)$$

Eigenschaften des Vektors c der komplexen Fourier-Koeffizienten

Bsp.: abfallender Sägezahn
Amplitude = 16, $N = 16$



$$\vec{c} = \frac{1}{N} \cdot \mathbf{F} \cdot \begin{pmatrix} 16 \\ 14 \\ 12 \\ 10 \\ 8 \\ 6 \\ 4 \\ 2 \\ 0 \\ -2 \\ -4 \\ -6 \\ -8 \\ -10 \\ -12 \\ -14 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 - 5,03j \\ 1 - 2,41j \\ 1 - 1,50j \\ 1 - j \\ 1 - 0,67j \\ 1 - 0,41j \\ 1 - 0,2j \\ 1 \\ 1 + 0,2j \\ 1 + 0,41j \\ 1 + 0,67j \\ 1 + j \\ 1 + 1,50j \\ 1 + 2,41j \\ 1 + 5,03j \end{pmatrix}$$

Wie in der 2. Zeile von (6) gezeigt, ist auch \vec{c} wie \mathbf{F} „spiegelsymmetrisch“ bezüglich der mittleren Komponente $c_{N/2}$ (sofern man die komplexen Komponenten von \vec{c} auch wieder als Polardiagramme darstellen würde), also $c_{(N-n)}^* = c_n$. Verwendet wurde dabei die komplexe Rechenregel: $(u \cdot v \cdot w)^* = u^* \cdot v^* \cdot w^*$ (u, v, w komplex) sowie $r^* = r$ für reelles r . Außerdem müssen c_0 und $c_{N/2}$ reell werden, da die entsprechenden Zeilen in \mathbf{F} nur aus reellen Zahlen bestehen ($F_{0k} = 1; F_{N/2,k} = \pm 1$). Sie heißen:

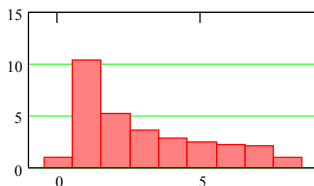
$$\begin{matrix} c_0 = a_0/2 & \text{DC-Anteil} \\ c_{N/2} & \text{Oberwellenrest} \end{matrix} \quad (7)$$

Alle Informationen stecken also bereits in $c_0 \dots c_{N/2}$. Der Rest des Vektors ist das konjugiert Komplexe der oberen Hälfte.

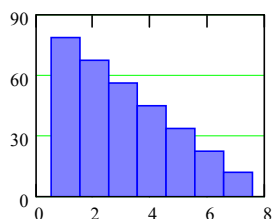
Der Index n gibt aber die Frequenz der zugehörigen Oberwelle an. Die fett gedruckte Erkenntnis spiegelt damit schon ein wichtiges Theorem in der Signaltechnik wieder: Wenn ich N äquidistante Punkte eines periodischen Signals kenne, so kann ich nur Amplituden und Phasen bis zur Oberwelle mit $N/2$ -facher Frequenz der Grundwelle bestimmen („Nyquist-Theorem“). Anwendung: ich will Musik bis 20kHz analysieren (= höchster hörbarer Ton), dann muss ich das Mikrofonsignal mindestens mit 40 kHz abtasten, also alle 25ms das Signal digitalisieren (ist bei hochwertigen Audio-wav-files auch ungefähr der Fall).

Gesamtamplituden und Phasen

Amplitudenspektrum
des obigen
Sägezahns



Phasenspektrum



Bei den reellen Fourierkoeffizienten gibt es die Möglichkeit, Gesamtamplituden A_n und Phasen φ_n anstelle von a_n und b_n zu verwenden:

$$\begin{aligned} f(x) &= \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx) \\ &= \frac{a_0}{2} + \sum_{n=1}^{\infty} A_n \cos(nx - \varphi_n) \end{aligned} \quad (8)$$

$$A_n = \sqrt{a_n^2 + b_n^2} \quad \varphi_n = \arctan \frac{b_n}{a_n}$$

Die letzte Zeile lässt sich aber auch leicht direkt aus den c_n bestimmen, wie ein Blick auf die Definition (3) zeigt. Zusammengefasst mit (7) ergibt sich schließlich:

$$\begin{aligned} A_n &= 2|c_n| & \varphi_n &= -\arg c_n & n &= 1 \dots \frac{N}{2} - 1 \\ A_0 &= c_0 & \text{DC-Anteil} & & A_{N/2} &= c_{N/2} & \text{Oberwellenrest} \end{aligned} \quad (9)$$

Für den Praktiker sind diese Größen meist interessanter als die a - und b -Koeffizienten. Sie ergeben sich gemäß (9) direkt und einfach aus den c 's und man kann aus ihnen sofort die wichtigen Amplituden- und Phasenspektren ableiten.

Inverse diskrete Fouriertransformation (IDFT): wie bekomme ich y_k wieder aus c_n ?

Beispiel N=4

$$\begin{array}{c|cccc} & 1 & 1 & 1 & 1 \\ & 1-j & -1 & j & \\ & 1 & -1 & 1 & -1 \\ & 1 & j & -1 & -j \\ \hline 1 & 1 & 1 & 1 & 4 & 0 & 0 & 0 \\ 1 & j & -1 & -j & 0 & 4 & 0 & 0 \\ 1 & -1 & 1 & -1 & 0 & 0 & 4 & 0 \\ 1 & -j & -1 & j & 0 & 0 & 0 & 4 \end{array}$$

Mithilfe der Inversen Matrix F^{-1} natürlich. Und die ergibt sich relativ einfach im Wesentlichen durch konjugiert komplex Bildung jedes Elementes von F :

$$F^* \cdot F = N \cdot 1 \tag{10}$$

wobei F^* aus den Elementen F_{nk}^* besteht. Beweis:

$$\begin{aligned} (F^* \cdot F)_{nk} &= \sum_{i=0}^{N-1} F_{ni}^* \cdot F_{ik} = \sum_{i=0}^{N-1} (w^*)^{n \cdot i} \cdot w^{i \cdot k} = \\ &= \sum_{i=0}^{N-1} \left(e^{j \frac{2\pi}{N}} \right)^{n \cdot i} \cdot \left(e^{-j \frac{2\pi}{N}} \right)^{i \cdot k} = \sum_{i=0}^{N-1} \left(e^{j \frac{2\pi}{N} (n-k)} \right)^i \end{aligned}$$

Auf der Diagonalen ist $n=k$ und die Summe besteht aus lauter Einsen. Für $n \neq k$ muss die Formel für geometrische Summen angewendet werden mit dem Resultat:

geometrische Summe:

$$\sum_{i=0}^{N-1} a^i = \frac{1-a^N}{1-a}$$

$$(F^* \cdot F)_{nk} = \sum_{i=0}^{N-1} \left(e^{j \frac{2\pi}{N} (n-k)} \right)^i = \frac{1 - e^{j \frac{2\pi}{N} (n-k) \cdot N}}{1 - e^{j \frac{2\pi}{N} (n-k)}} = \frac{1 - e^{j 2\pi (n-k)}}{1 - e^{j \frac{2\pi}{N} (n-k)}} = 0$$

zusammengefasst :

$$(F^* \cdot F)_{nk} = \begin{cases} N & \text{für } n = k \\ 0 & \text{für } n \neq k \end{cases}$$

Mit diesen Resultaten wollen wir Gleichung (5) von links mit F^* multiplizieren und erhalten:

$$F^* \cdot \vec{c} = F^* \cdot \frac{1}{N} \cdot F \cdot \vec{y} = \frac{1}{N} \cdot F^* \cdot F \cdot \vec{y} = \frac{N}{N} \cdot \vec{y} = \vec{y}$$

$$\vec{y} = F^* \cdot \vec{c} \quad \text{oder :}$$

$$y_k = \sum_{n=0}^{N-1} (w^*)^{n \cdot k} \cdot c_n = \sum_{n=0}^{N-1} e^{j \frac{2\pi}{N} \cdot n \cdot k} \cdot c_n = \tag{11}$$

$$= c_0 + \sum_{n=1}^{\frac{N}{2}-1} \left(e^{j \frac{2\pi}{N} \cdot n \cdot k} \cdot c_n + e^{j \frac{2\pi}{N} \cdot (N-n) \cdot k} \cdot c_n^* \right) + e^{j \frac{2\pi}{N} \cdot \frac{N}{2} \cdot k} \cdot c_{N/2}$$

In der letzten Zeile wurde die Summe in 2 Hälften aufgespalten (z.B. N=16: 1...7 und 15...9) und die Symmetrie (6) der c_n wurde ausgenutzt (z.B. $c_3 = c_{13}^*$ oder $c_{13} = c_3^*$). (11) kann weiter vereinfacht werden:

$$\begin{aligned} e^{j \frac{2\pi}{N} \cdot (N-n) \cdot k} &= e^{j \left(2\pi k - \frac{2\pi}{N} \cdot n \cdot k \right)} = e^{j 2\pi k} \cdot e^{-j \frac{2\pi}{N} \cdot n \cdot k} \stackrel{\text{wg(4)}}{=} 1 \cdot e^{-j \cdot x_k \cdot n} \\ e^{j \frac{2\pi}{N} \cdot n \cdot k} \cdot c_n + e^{j \frac{2\pi}{N} \cdot (N-n) \cdot k} \cdot c_n^* &\stackrel{\text{wg(4)}}{=} c_n \cdot e^{j \cdot x_k \cdot n} + c_n^* \cdot e^{-j \cdot x_k \cdot n} = \\ &\stackrel{\text{wg(3)}}{=} \frac{a_n - j b_n}{2} \cdot e^{j \cdot x_k \cdot n} + \frac{a_n + j b_n}{2} \cdot e^{-j \cdot x_k \cdot n} = \\ &= a_n \frac{e^{j \cdot x_k \cdot n} + e^{-j \cdot x_k \cdot n}}{2} + b_n \frac{e^{j \cdot x_k \cdot n} - e^{-j \cdot x_k \cdot n}}{2j} = a_n \cos nx_k + b_n \sin nx_k \\ e^{j \frac{2\pi}{N} \cdot \frac{N}{2} \cdot k} &= e^{j \cdot \pi \cdot k} = (-1)^k = \cos \pi k = \cos \left(\frac{N}{2} \cdot \frac{2\pi}{N} \cdot \pi k \right) \stackrel{\text{wg(4)}}{=} \cos \left(\frac{N}{2} \cdot x_k \right) \end{aligned}$$

Trigonometrische Interpolation

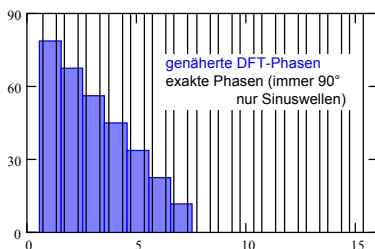
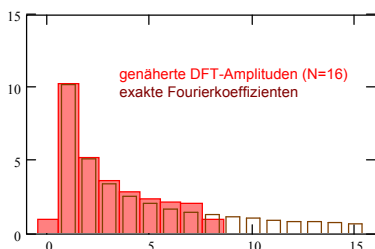
Setzt man die Vereinfachungen in (11) ein, so erhält man die Formel für die sog. trigonometrische Interpolation:

$$y_k = c_0 + \sum_{n=1}^{\frac{N}{2}-1} (a_n \cos nx_k + b_n \sin nx_k) + c_{N/2} \cdot \cos\left(\frac{N}{2} \cdot x_k\right) \tag{12}$$

$$\stackrel{\text{wg(8)}}{=} A_0 + \sum_{n=1}^{\frac{N}{2}-1} A_n \cos(nx_k - \varphi_n) + A_{N/2} \cdot \cos\left(\frac{N}{2} \cdot x_k\right)$$

Mithilfe der aus der DFT gewonnenen Koeffizienten c_n oder (a_n, b_n) oder (A_n, φ_n) kann man also Grundwelle $\cos x$ und Oberwellen $\cos nx$ so zu einer Gesamtfunktion $g(x)$ überlagern, dass $g(x)$ exakt durch alle vorgegebenen Wertepaare (x_k, y_k) durchgeht. Solche Funktionen nennt man Interpolationsfunktionen.

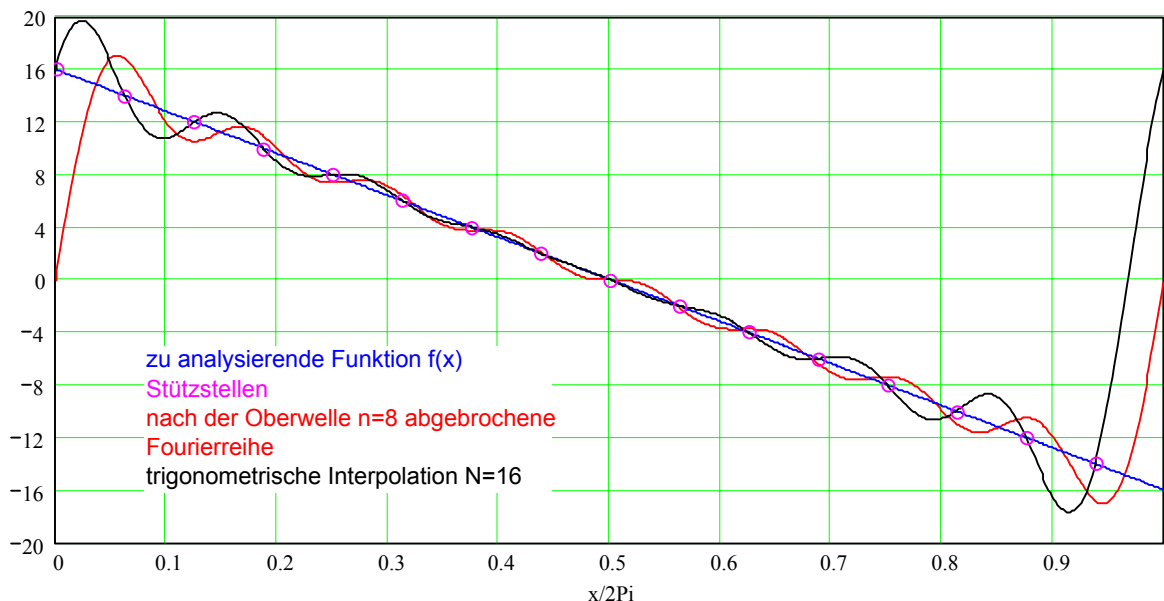
Beispiel: Sägezahn (N=16)



$$g(x) := A_0 + \sum_{n=1}^{\frac{N}{2}-1} A_n \cos(nx - \varphi_n) + A_{N/2} \cos\left(\frac{N}{2} \cdot x\right)$$

$$\stackrel{\text{wg(12)}}{=} y_k$$

Man beachte den Unterschied zur Fourierreihe: diese konvergiert gegen die zu analysierende Funktion $f(x)$ (meistens) im gesamten Bereich $0 \leq x \leq 2\pi$. Bricht man die Fourierreihe zur Fouriersumme ab, so erhält man eine Näherung, die zwar nahe bei $f(x)$ liegt, mit dieser aber nur nicht vorhersehbare Schnittpunkte gemeinsam hat. Die trigonometrische Interpolation $g(x)$ dagegen geht durch alle Stützstellen (x_k, y_k) exakt durch, die den Dateninput der DFT bildeten. Für große N spielen diese Unterschiede allerdings keine Rolle mehr, die Amplituden und Phasen aus der DFT stimmen praktisch mit den exakten Fourierkoeffizienten (2) überein, wenn man Beispiele heranzieht, bei denen man (2) auch analytisch exakt bestimmen kann.



Divide et Impera: Fast Fourier Transform (FFT)

Teile und herrsche, dann fällt das herrschen leichter, meinten schon die Römer. Gauß erkannte 1805 die Möglichkeit der Arbeitersparnis bei der Berechnung der DFT, Cooley und Tukey veröffentlichten dann 1965 den legendären Algorithmus, der ihren Namen trägt. Sie spezialisierten sich auf den in der Praxis häufig auftretenden und schon anfangs erwähnten Spezialfall $N=2^m$. Die Berechnung der

Matrixmultiplikation $\vec{c} = \frac{1}{N} \cdot \mathbf{F} \cdot \vec{y}$ in (5) erfordert grob N^2 komplexe

Multiplikationen und Additionen, also je über 1Mio. Operationen bei $N=1024$. Man kann die $\mathbf{F}(N=1024)$ -Multiplikation aber auf die $\mathbf{F}(N=512)$ -Multiplikation zurückführen, wie die folgende Rechnung zeigt. Dies spart ungeheuer! Im weiteren wird bei der Matrix \mathbf{F} und beim Schnörkelfaktor w in Klammern dahinter geschrieben, für welche Einteilung N die Größen gelten. Also $\mathbf{F}(1024)$ ist eine 1024×1024 -

Matrix, $w(1024) = e^{-j \cdot \frac{2\pi}{1024}}$.

Der Trick besteht lediglich im Zusammenfassen der geraden Indizes k und der ungeraden in (4):

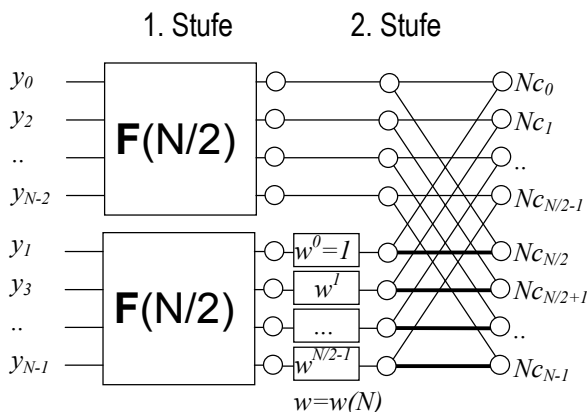
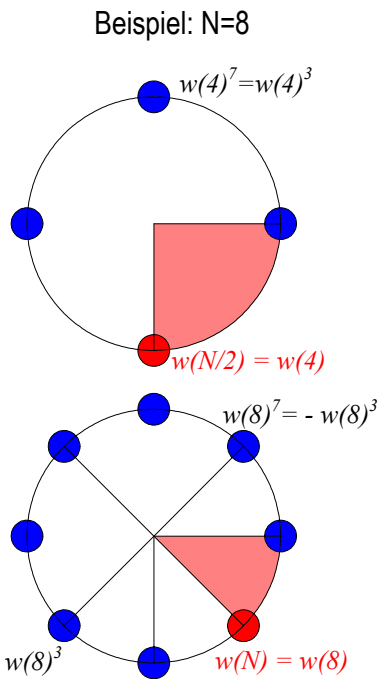
$$\begin{aligned}
 Nc_n &= \sum_{k=0}^{N-1} e^{-j \frac{2\pi}{N} n \cdot k} \cdot y_k = \sum_{m=0}^{N/2-1} \left(e^{-j \frac{2\pi}{N} n \cdot 2m} \cdot y_{2m} + e^{-j \frac{2\pi}{N} n \cdot (2m+1)} \cdot y_{2m+1} \right) \\
 &= \sum_{m=0}^{N/2-1} e^{-j \frac{2\pi}{N/2} n \cdot m} \cdot y_{2m} + e^{-j \frac{2\pi}{N} n} \cdot \sum_{m=0}^{N/2-1} e^{-j \frac{2\pi}{N/2} n \cdot m} \cdot y_{2m+1} = \\
 &= \sum_{m=0}^{N/2-1} w(N/2)^{n \cdot m} \cdot y_{2m} + w(N)^n \cdot \sum_{m=0}^{N/2-1} w(N/2)^{n \cdot m} \cdot y_{2m+1}
 \end{aligned}$$

In der letzten Zeile stehen aber im Wesentlichen zwei $N/2$ -DFT's, jedenfalls solange $n < N/2$. Für $n \geq N/2$ ist die Periodizität

$$w(N/2)^{\frac{N}{2}+n} = w(N/2)^n \text{ sowie } w(N)^{\frac{N}{2}+n} = -w(N)^n \text{ zu}$$

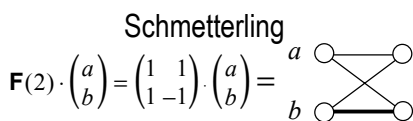
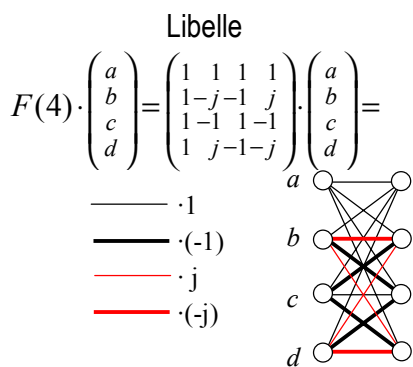
berücksichtigen (s. Torten links). Man kann dann n auf den Bereich $0 \leq n \leq \frac{N}{2} - 1$ einschränken und schreiben:

$$\left. \begin{aligned} Nc_n \\ Nc_{\frac{N}{2}+n} \end{aligned} \right\} = \underbrace{\sum_{m=0}^{N/2-1} w(N/2)^{n \cdot m} \cdot y_{2m}}_{N/2\text{-DFT}} \underbrace{\left\{ \begin{aligned} + \\ - \end{aligned} \right\}}_{\text{2. Stufe: gemischte Addition}} \underbrace{\sum_{m=0}^{N/2-1} w(N/2)^{n \cdot m} \cdot y_{2m+1}}_{N/2\text{-DFT}} \quad (13)$$



Damit ist das Ziel erreicht. Man erhält eine 2-stufige DFT. In der 1. Stufe wird mit den geraden y_{2m} und den ungeraden y_{2m+1} je eine $N/2$ -DFT durchgeführt. In der 2. Stufe erfolgt die gemischte Addition gemäß (13). Am besten macht man sich das Verfahren in einem Diagramm deutlich. In der 2. Stufe bedeuten:
Dünner Strich = Weiterleitung der Zahl nach rechts
Dicker Strich = Multiplikation mit -1 und Weiterleitung
Kasten = Multiplikation mit Zahl im Kasten und weiter
Alle von links in einen Knoten einlaufende Striche werden addiert

Mehrstufige FFT's : Libellen und Schmetterlinge

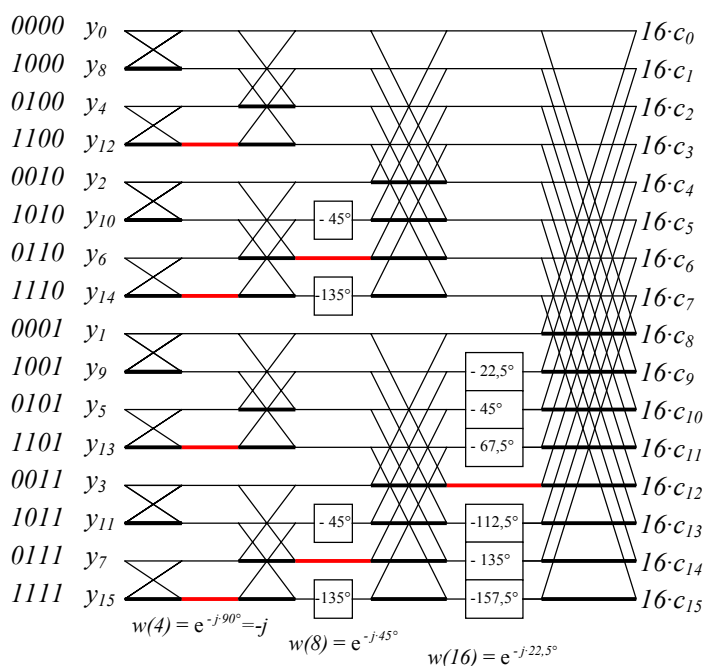


Schema 13 halbiert ungefähr den Rechenaufwand bei großen DFT's (z.B. N=1024). Es liegt natürlich nahe, die F(N/2)-Multiplikation nach demselben Schema in F(N/4)-Multiplikationen aufzuspalten usw. Zu beachten ist dabei, dass bei jeder neuen Stufe die Eingangswerte y_k umgeordnet werden müssen. Im 3-stufigen Fall wäre die Reihenfolge von oben nach unten z.B.:

$y_0, y_4, y_8 \dots y_2, y_6, y_{10} \dots y_1, y_5, y_9 \dots y_3, y_7, y_{11} \dots$

Beim betrachteten Spezialfall $N=2^m$ gelangt man so nach m-1 Stufen zur F(4) und nach m Stufen zur F(2). Mit dem bei (13) eingeführten Diagrammstil kann man F(4) als „Libelle“ (dragonfly) und F(2) als „Schmetterling“ (butterfly) darstellen.

Bei jeder neuen Stufe, die auf der linken Seite dazu kommt, müssen die zugehörigen Schnörkelfaktoren w bestimmt werden. Ihre Winkel verdoppeln sich jeweils von rechts nach links. So bekommt man für N=16 letztendlich folgendes völlig in Schmetterlinge aufgelöste Schema:



Die dicken schwarzen Striche bedeuten wieder $\cdot(-1)$, die dicken roten $\cdot(-j)$. Die Knoten wurden der Übersichtlichkeit halber weggelassen, bei den Potenzen der Schnörkelfaktoren stehen aus demselben Grund nur die Phasen in den Kästchen. Man sieht, dass bei kompletter Zerlegung runter bis F(2) die Eingangswerte letztlich in sog. „bit reverse order“ angeordnet werden müssen, die Binärzähler für die Indizes also von links nach rechts arbeiten müssen. Der Deutlichkeit halber ist auf der linken Seite deshalb auch noch das Bitmuster der Indizes mit aufgeführt.

Man hat ausgerechnet, dass bei völliger Zerlegung in Schmetterlinge die 1024-Punkte-DFT nur noch 0,4% der ursprünglichen Multiplikationen und nur noch 1% der ursprünglichen Additionen benötigt!